Package 'CPE'

September 26, 2018

Title Concordance Probability Estimates in Survival Analysis

Version 1.5.1

Depends R (>= 2.10.0),survival,rms

Author Qianxing Mo, Mithat Gonen and Glenn Heller

Maintainer Qianxing Mo <qianxing.mo@moffitt.org>

Description Functions to calculate concordance probability estimates in survival analysis.

LazyData no

License GPL (>= 2)

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-09-25 23:00:07 UTC

R topics documented:

---

phcpe *Gonen \& Heller Concordance Probability Estimate for the Cox Proportional Hazards model*

Description

A function to calculate Gonen \& Heller concordance probability estimate (CPE) for the Cox proportional hazards model.

Usage

```
phcpe(coxfit, CPE.SE=FALSE,out.ties=FALSE)
```

Arguments

coxfit	A coxph or cph object
CPE.SE	A logical value indicating whether the standard error of the CPE should be calculated
out.ties	If out.ties is set to FALSE,pairs of observations tied on covariates will be used to calculate the CPE. Otherwise, they will not be used.

Value

CPE	Concordance Probability Estimate
CPE.SE	the Standard Error of the Concordance Probability Estimate

Author(s)

Qianxing Mo, Mithat Gonen and Glenn Heller; <qianxing.mo@moffitt.org>

References

Mithat Gonen and Glenn Heller. (2005). Concordance probability and discriminatory power in proportional hazards regression. *Biometrika*, 92, 4, pp.965-970 Glenn Heller and Qianxing Mo. (2016). Estimating the concordance probability in a survival analysis with a discrete number of risk groups. *Lifetime Data Analysis*, 22(2):263-79.

See Also

phcpe2

Examples

```
### create a simple data set for testing
set.seed(199)
nn <- 1000
time <- rexp(nn)
status <- sample(0:1, nn, replace=TRUE)
covar <- matrix(rnorm(3*nn), ncol=3)
survd <- data.frame(time, status, covar)
names(survd) <- c("time","status","x1","x2","x3")

coxph.fit <- coxph(Surv(time,status)~x1+x2+x3,data=survd)

### Calculate CPE only (needs much less time).
phcpe(coxph.fit)
phcpe(coxph.fit,out.ties=TRUE)
#result is identical because the covariates are not tied #

### Calculate CPE and CPE.SE
phcpe(coxph.fit, CPE.SE=TRUE)
phcpe(coxph.fit, CPE.SE=TRUE,out.ties=TRUE)
```

```
#*** For unknown reason, 'coxph.fit' may need to be removed before running cph()***
rm(coxph.fit)

cph.fit <- cph(Surv(time, status)~x1+x2+x3, data=survd,method="breslow")

### Calculate CPE only (needs much less time).
phcpe(cph.fit)
phcpe(cph.fit,out.ties=TRUE)

### Calculate CPE and CPE.SE
phcpe(cph.fit, CPE.SE=TRUE)
phcpe(cph.fit, CPE.SE=TRUE,out.ties=TRUE)
```

---

phcpe2 *Gonen \& Heller Concordance Probability Estimate for the Cox Proportional Hazards model*

Description

A function to calculate Gonen \& Heller concordance probability estimate (CPE) for the Cox proportional hazards model.

Usage

```
phcpe2(coef,coef.var,design, CPE.SE=FALSE,out.ties=FALSE)
```

Arguments

coef	The coefficients of the Cox model.
coef.var	The covariance matrix of the coefficients of the Cox model.
design	A design matrix for covariates. The rows correspond to subjects, and the columns correspond to covariates.
CPE.SE	A logical value indicating whether the standard error of the CPE should be calculated
out.ties	If out.ties is set to FALSE,pairs of observations tied on covariates will be used to calculate the CPE. Otherwise, they will not be used.

Value

CPE	Concordance Probability Estimate
CPE.SE	the Standard Error of the Concordance Probability Estimate

Author(s)

Qianxing Mo, Mithat Gonen and Glenn Heller; <qianxing.mo@moffitt.org>

References

Mithat Gonen and Glenn Heller. (2005). Concordance probability and discriminatory power in proportional hazards regression. *Biometrika*, 92, 4, pp.965-970 Glenn Heller and Qianxing Mo. (2016). Estimating the concordance probability in a survival analysis with a discrete number of risk groups. *Lifetime Data Analysis*, 22(2):263-79.

See Also

phcpe

Examples

```
### create a simple data set for testing
set.seed(199)
nn <- 1000
time <- rexp(nn)
status <- sample(0:1, nn, replace=TRUE)
covar <- matrix(rnorm(3*nn), ncol=3)
survd <- data.frame(time, status, covar)
names(survd) <- c("time","status","x1","x2","x3")

coxph.fit <- coxph(Surv(time,status)~x1+x2+x3,data=survd)

phcpe(coxph.fit,CPE.SE=TRUE)
phcpe2(coef=coxph.fit$coefficients,coef.var=coxph.fit$var,design=model.matrix(coxph.fit))

#*** For unknown reason, 'coxph.fit' may need to be removed before running cph()***
rm(coxph.fit)

cph.fit <- cph(Surv(time, status)~x1+x2+x3, data=survd,method="breslow")

### Calculate CPE only (needs much less time).
phcpe2(cph.fit$coefficients,coef.var=cph.fit$var,design=model.matrix(cph.fit),CPE.SE=TRUE)
```

Index